2020

Automated analysis and indexing of lecture videos

Gayathri Sreepathy
Iowa State University

Automated analysis and indexing of lecture videos

by

Gayathri Sreepathy

A thesis submitted to the graduate faculty in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Major: COMPUTER SCIENCE

Program of Study Committee:
Simanta Mitra, Major Professor
Gurpur Prabhu
Carl Chang

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this thesis. The Graduate College will ensure this thesis is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2020

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

ACKNOWLEDGMENTS

I would like to take this opportunity to express my thanks to those who helped me with aspects of conducting research and writing of this report. I would like to express my gratitude to Dr. Simanta Mitra for his guidance and support throughout the course of this research, and to Dr. Gurpur Prabhu for guiding me on how to conduct research. I would also like to thank my committee member Dr. Carl Chang for his help and support.

ABSTRACT

Learning from online videos mainly helps the students and every individual understand a specific topic easily because of the realistic picturization. One of resources available to students is automated analysis and indexing of online lecture videos using image processing. Many online educational organizations and universities use video lectures to support teaching and learning. In past decades, video lecture portals have been widely used and are very popular. The text displayed in these video lectures are a valuable source for analyzing and indexing the lecture contents. Considering this scenario, we present an approach for automatic analysis and indexing of lecture videos using OCR (Optical Character Recognition) technology. For this, we segregated the unique key frames from a lecture video to extract the video contents. After the segregation of key frames by applying OCR and ASR (Automatic Speech Recognition) technology we can extract the textual data contents from the video lecture. From the obtained metadata, we segmented the video lecture based on the time-based text occurrence of the topics. The performance and the effectiveness of proposed analysis and indexing is proven by the evaluation.

CHAPTER 1. INTRODUCTION

Students achieve learning goals from many resources like university lecture videos, supplemental extra material, and many online video platforms. Since there is a rise in the amount of lecture videos that are available today, students find it difficult to look for a desired topic. Every time the student must investigate an entire video to find a specific topic that they really need. Even though a student has found the useful video, it is still difficult for the student to find whether the video is useful by scanning the title or other keywords in the video. Hence the problem becomes how to find all the possible information in a pool of videos. Online video platforms like YouTube, Coursera, Udemy, Lynda , etc. retrieve and search videos based on the available text that is stated as keywords, title, and brief description. Generally, this information is created by the human to ensure proper retrieval of the video list when the user searches for it, which is more time and cost consuming. Furthermore, this information represents only the brief description and at a high level. Therefore, the next generation of technology automatically captures the metadata from the video by using image processing techniques. The free availability of online lecture videos helps students [1], improve their goal [2] and achieve their insight [3]. Moreover, the accessibility of video lectures can help the different pedagogy settings [4] and overturned lecture hall [5]. Unluckily, the creation of high feature “educational pills” such as TED approach [6] is costly for colleges. The usage of MOOC’s [7] [8] is preferable, if the video lectures guarantee a large group of students and has a stability of content. Most of the universities these days prefer low cost solutions for video indexing from the live class recordings. The features of many universities where the professor lecture videos are taped and

then they are shared with the students with least amount of processing efforts have been described in research ([9] [10] [11] [12] [13] [14]). The main value of our solution is its cost effectiveness so that students can benefit from the lecture videos to maintain a connection between in-class and out-of-class learning. Related to “video pills”, the video recordings have a main difficulty: their length is usually same as the in-class video recordings, therefore it makes difficult for the student to find the specific content that is presented by the professor, among other activities and exercises. In [16] the authors have focused on creating bookmarks during lecture which helped to improve topic search and then make it available to the students. But this method requires the contribution of students and professor which is likely to fail if they are not contributing. The main challenge is to perform video indexing that should be readily available to the students so they can search the topics that are required. In [17] [18] [19] Video metadata analysis and indexing are the main focus. Video indexing can be done by analyzing audio, video and textual information that appears on the blackboard. In [35] the authors have focused on the speech transcription of the professor. Analyzing the speech transcription is difficult because the talker does not follow proper grammar and non-native speakers have different way of pronunciation which become challenging for analysis. Therefore, they used a knowledge base for properly annotating the video lecture which is effective for multilingual and nonnative speakers. In previous methodologies, the authors have extracted metadata from slide, professor notes and used it for the video search. But there are many more source of metadata that can be used for the video search and indexing. It includes slides, the text that appears on the video (can be any software or pictures within the video) and the speech transcription of the professor. In our approach we consider all the possible source of metadata for the purpose of video indexing and

video search. As stated before, extracting metadata data from content-based technique is an important area of research. In this research, our main problem is to find an effective way to automatically extract metadata from the video lecture and use it for the purpose of video search and indexing. Our research purpose can be formulated as a) Can the retrieved metadata help the learner in searching the video contents effectively? b) Can these retrieved metadata also be used for the purpose of indexing where users can easily navigate to a particular part of the video?

This research is structured as follows:

In Chapter 2, we explain the concepts of OCR, ASR, Video search and discuss the previous work done to find the video contents.

In Chapter 3, we discuss our methodology applied to extract metadata using these algorithms.

In Chapter 4, we discuss and evaluate our results.

In Chapter 5, we conclude the results and outcomes of the research, and discuss future work that can be taken as an extension of this research.

CHAPTER 2. RELATED WORK

Video search, indexing, metadata analysis and navigation to certain part of video are some of the challenging areas of research. In [17], they have focused on video segmentation, annotation, and recommendation. Taking those into consideration, we will analyze the metadata from certain parts of the video for the purpose of indexing.

This section provides a summary of past research work done in the domain of video analysis. We have classified the research purpose based on their outcomes and focus of research area. Our research purpose is to classify video lectures based on the extracted metadata, transcript, and image processing. The student should be able to search a specific topic and navigate to it instead of watching the entire video. The student can also look for specific concept and exercise that was discussed by the professor. In [17][28][24][23], the authors have differently focused on their research. In [28], the authors have focused on speech transcription using ASR and vocabulary content matching. In [24] the authors have combined text from the slides and professor speech, performed video indexing and video search using OCR & ASR. In [23] the authors have annotated the video lecture with information retrieved for multilingual audios. But other research has focused differently in these areas. For example, paper [25] has extracted the text from the slides. In [26] the authors have presented video tags, but in [27] they have extracted keywords.

Video segmentation is the process of segmenting or extracting few parts of the video lecture over a specific topic which is also another challenging research area. In [29] [30] [31] [32] [33] [34] the authors have focused on video segmentation based on user query. Next, we

have categorized the works based on the models used to obtain metadata. The key source of metadata in a video recording are from the audio of the professor and text that appears on the video. The papers in [17] [25] [29] focus on analysis of text that appears on the video. Research [26] [27] [28] [30] [33] [34] examined the audio of the professor and[23] [24] [32] they combine the source of available information. The methods established on the evaluation of visual metadata ([17] [25] [29] uses OCR algorithm, to detect the slide transition.

In [25] the research focus is on text recognition and identification of slide areas, whereas in [17] the goal was to perform video search using dual frame. The methodology offered in [29] identifies slide labels and uses them for the purpose of video segmentation. Other than OCR technology, [31] utilized a histogram technique to find the video transition between each frame. From the above-stated methodologies, our work uses OCR algorithm and a background subtraction methodology to detect slide transition and automatically find unique key frames for the purpose of video segmentation. Our one of the main focus is to analyze each unique key frame and extract topic and content of each slide that finally constitute the metadata.

Besides analyzing the voice transcript, the researchers of [26] completed a text search, whereas in [27] they focused on label extraction using text mining procedures. The method presented in [30] utilizes Wikipedia to evaluate text similarities for video segmentation. The methodology presented in [28] uses a terminology extraction from the professor material to understand where the topics are found in the lecture video and further used for the purpose of video indexing. Finally, in [33] and [34] the authors follow graph partitioning methodology to implement video segmentation. The retrieval of video lecture system described in [23] [24]

combines metadata that are extracted from slides and professor speech. Likewise, this research also focuses on video lecture indexing using speech transcription metadata.

CHAPTER 3. METHODOLOGY

This section describes about the structure of lecture videos, purpose of unique key frames, video indexing methods and video search methods. We present an approach for automatic analysis, search and indexing of lecture videos using OCR and ASR technology. For the pre-processing step required for feature-based techniques, we extract the images from a lecture video using FFMPEG library. The features extracted from the video that includes all images in the video are used for collecting the metadata. The metadata includes the image file and the corresponding frame number.

3.1 Video Structure

The main sources for lecture videos are from our university online lecture videos database captured during a live session. Most of the lecture videos use multi scenes format (e.g., Figure 3.1) in which the lecturer and the lecture slides are displayed synchronously. The basic format of lecture videos consists of 2 main parts: the lecture videos are recorded by using a video camera and captures the desktop of the professor's computer. For indexing and search, no synchronization is required between the video and the lecture slides because our focus is on the slides which is the main requirement. Our research mainly focuses on the lecture videos which has two screens (professor & slides) because two videos are coordinated during the recording process. Hence, complete unique slides can be obtained which are captured from the lecturer desktop screen. The extracted slide frames can help us to retrieve the video contents. There are 3 main sources of metadata that we can extract to perform video indexing and search:

a) Text that appears on the slide.

b) Voice transcription of the professor.

c) White board contents.

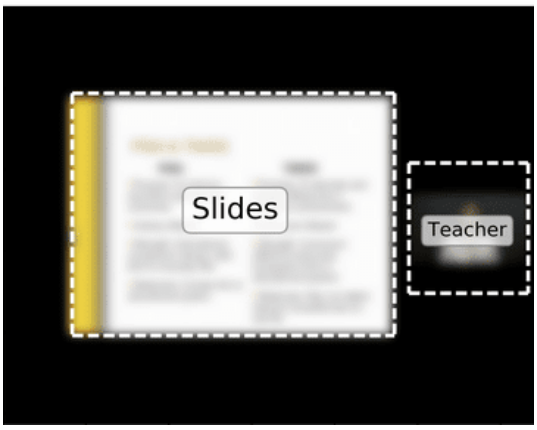

Figure 3.1 An example for lecture video format

3.2 Unique frame detection

Each lecture video can be segmented into images by extracting 30 frames per second. A lot can happen in each second, hence capturing 30 frames per second is highly recommended based on the video quality. After 30 frames are generated per second, we apply background subtraction algorithm to find the difference (pixel value) between each frame. The difference value constitutes the image threshold value. If the value of image threshold is greater than 30% then the 2 images are different and constitute the unique key frames.

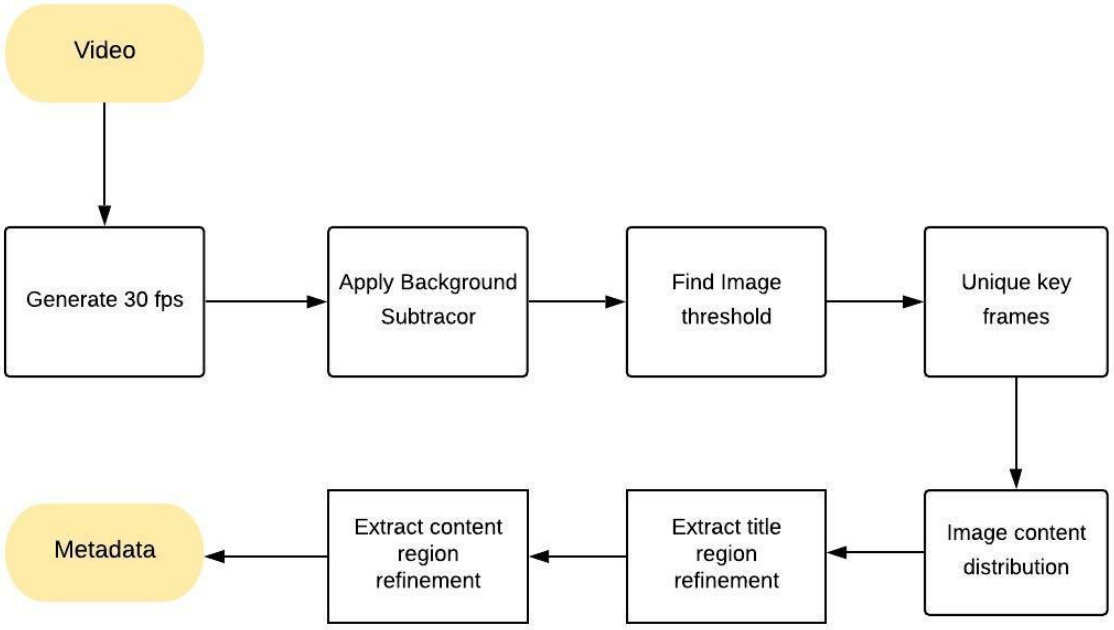

Figure 3.2 Unique frame detection approach

For example, consider a frame without mouse pointer and another frame with mouse pointer, the image threshold would be lesser than 30%. Hence, we apply threshold to ignore such exceptions. If the threshold of two frames are greater than 30% then it represents as key frames or else, it is discarded. After we have collected all possible unique frames, we use tesseract library to extract all the text from each frame which is collected as metadata and then used for the purpose of video analysis. A frame or a slide has mainly two parts, 1) Title Region Refinement and 2) Content Region Refinement. Therefore, these titles can then be used to segment the video and can also act as video tags. Both title and content metadata of each slide are used for video search.

We use background subtraction method to find unique key frames.(Figure 3.2) If the unique frame in the video lecture does not have any change for a given amount of time, then we know the video frame change has ended, in which we can capture the current frame. A background subtractor model is initialized and is applied to every frame. Three variables are initialized a) Captured: A Boolean variable which indicates whether a frame has been captured. b) Total: A counter which indicates how many frames have captured and c) Frames: A counter which indicated how many frames are processed.

The collected video frames are then looped so that it will be easy to grab the unique frames. The background subtractor is applied whenever the video frames are looped producing our mask. Black pixels represent the background of the frame and white pixels represent the result of difference between two frames. Then we apply a series of structural operations to remove the noise in the video loop. We then calculate the percentage of the mask that is "foreground" versus "background". Compare the foreground pixel percentage to the min_percent constant. If the pixel percentage is less than min_percent of the frame has motion, then the frame is not captured. Otherwise we will save this frame to the disk. The unique images are then used to extract text and other metadata from the images.

3.3 Video Indexing & Search Approach

Similar to textbook index, students and individuals can benefit from video lecture if the videos are indexed and can easily navigate to the specific part of the video [17]. The new emerging technology video analysis has gained a lot of attention from researchers due to increase

access to online video lectures. Education techniques are becoming more and more transportable and adaptable supporting successful indexing approach which can be beneficial for students [18].

After we have unique frames from background subtraction method, we apply tesseract to extract textual information from the slides. Another method to extract metadata is by applying ASR methods. The speech transcription is extracted using Microsoft azure technology because of cost efficiency and easy transportability. To obtain video search we need text from the video and audio of the professor so students can even look up for specific topics from the voice. Figure 3.3 explains the methods that are performed in this research.

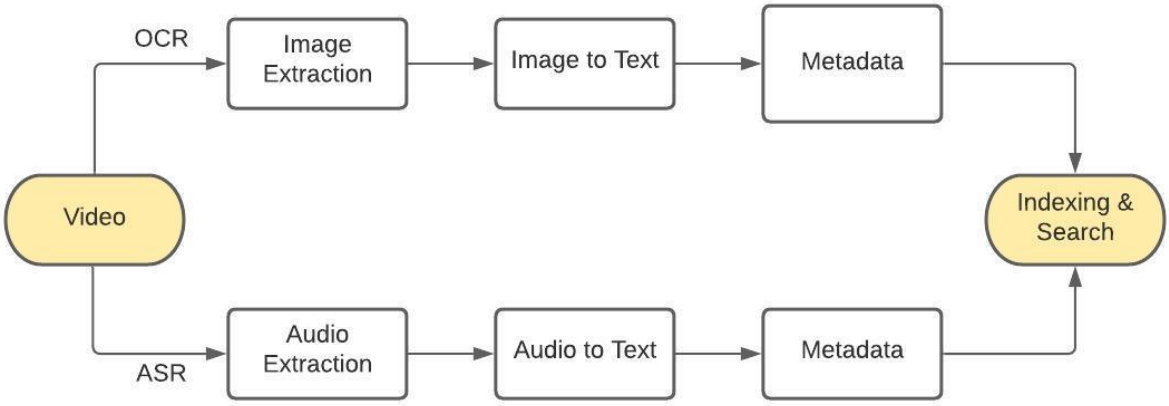

Figure 3.3 Video indexing and search approach

CHAPTER 4. EXPERIMENT AND RESULTS

This section describes the experiments used for analysis and indexing of a video lecture. This research focuses on analyzing a video lecture either from the professor or university's live recording and then make it beneficial for students so that they can search for a specific topic. We follow the below approach for the purpose of video indexing, analysis, and search.

- The OCR technology uses tesseract library to extract text from the video.
- Microsoft azure technology to transcribe the voice of the professor.

4.1 OCR

We use OpenCV library to identify slide transition and extract the unique key frames. Background subtraction is one of the major steps to find unique key frames. This method is used to extract the moving slides from a static environment. Image processing is used to process, analyze, and use images in order to extract metadata that can be used for video indexing. Some of the common image processing tasks include displaying images, clipping, flipping, rotating, etc. Here are the image processing approaches that we used in our work using only threshold and contour filtering:

4.1.1 Obtain Binary Image

First, we load the unique images that we extracted from the background subtractor method. Then each image is converted into grayscale because it helps to eliminate the noise, easy to code and efficient to process the image in gray scale than color image. Figure 4.1 is an example for a grayscale image.

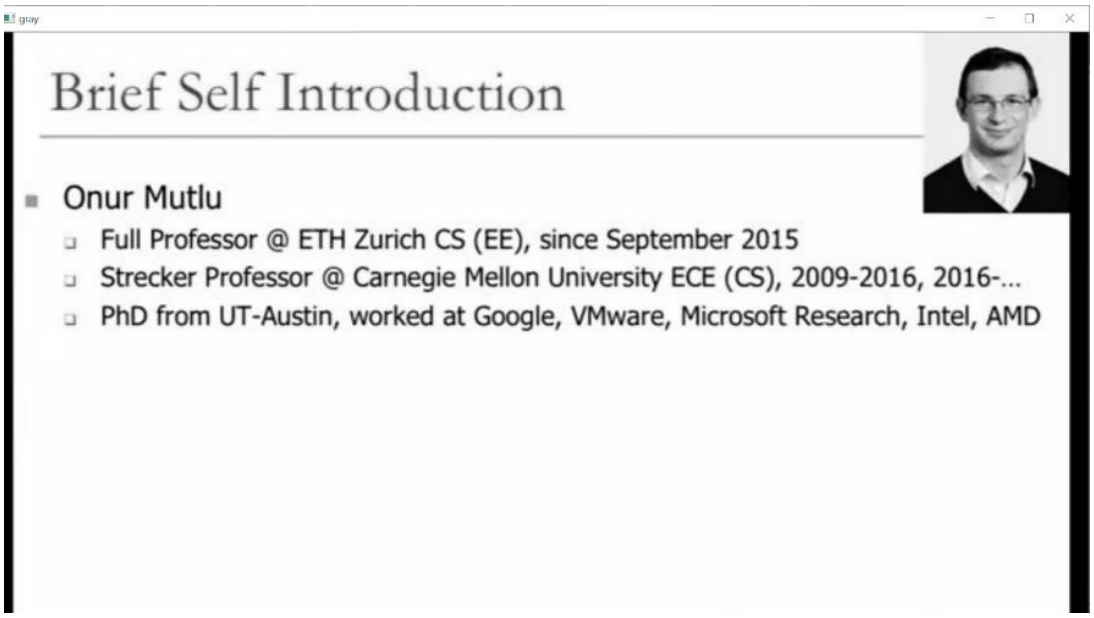

Figure 4.1 Example for grayscale image

Next, we apply image blurring useful to remove noise. It removes high frequency content from the image resulting in edges being blurred when a loss-pass Gaussian kernel filter is applied. It is done with the function **cv2.GaussianBlur().** We define the width and height of the kernel. We also specified the standard deviation in the X and Y direction, SigmaX and SigmaY respectively. Figure 4.2 is the resulting image after applying Gaussian Blur. Gaussian filtering is highly effective in removing Gaussian noise from the image.

Figure 4.2 After applying Gaussian Blurring to grayscale image

After Gaussian blurring, we apply an adaptive threshold algorithm which calculates the value of threshold for the given image. Therefore, we get different values of threshold for different areas of the image and helps us to view better results for images with different values . It is done with the function **cv2.adaptiveThreshold()** which decides how the thresholding value is calculated based on these 2 parameters - cv2.adaptive_thresh_mean_c and cv2.adaprive_thresh_gaussian_c where c is just a constant. Figure 4.3 is the image after applying adaptive threshold.

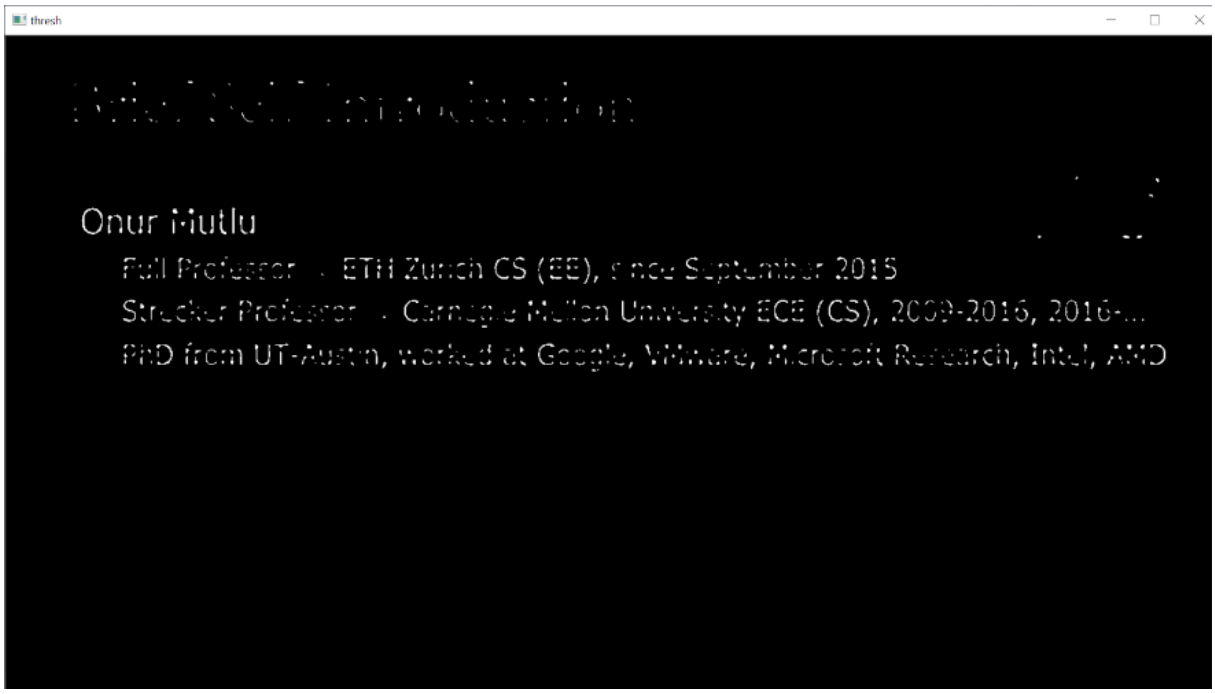

Figure 4.3 After finding adaptive threshold we obtain binary image

4.1.2 Combine adjacent text

After obtaining the binary image from the previous step we apply structuring elements to the text area and then dilate to form a single contour. Structuring elements is an array of 0's and 1's that identifies the pixel in the image being processed. It also identifies the neighboring pixel used for processing. It is done with the function **cv2.GetStructuring Elements().** We specify the desired shape like elliptical/circular shaped, cross shaped and rectangular shaped. We have used a rectangular shaped kernel for finding the pixel that has some text and the size of the kernel.

Next, we dilate the image to combine the text into a single contour. In a binary image, if the value of the pixel is 1 and if their neighboring pixel value is 1 then those pixels are

selected and outputted. It uses the function **cv2.dilate().** We specify the image and the kernel output from the rectangular structuring elements(Figure 4.4). The main purpose of morphological dilation is to make objects more visible and fill in small holes in objects.

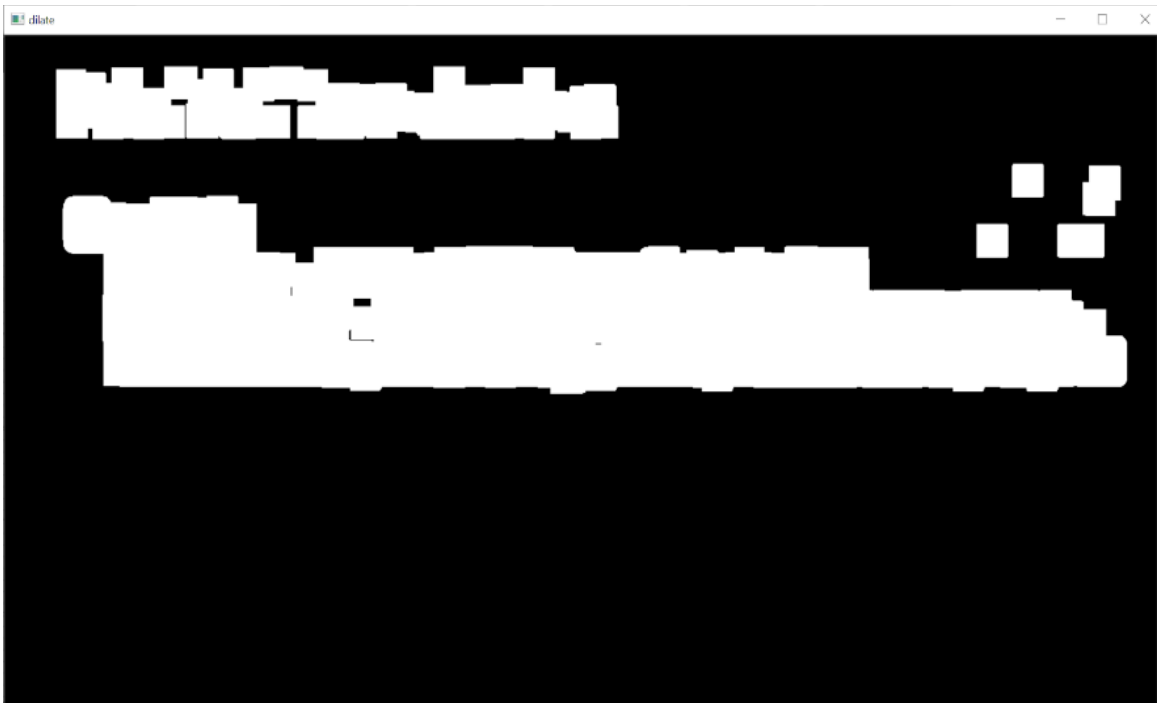

Figure 4.4 After rectangular structuring kernel and dilation

4.1.3 Filter for text contours

After obtaining the binary image from the previous step we find contours and filter using the contour area. From here we can draw the bounding box with **cv2.rectangle().** Contours help us to find all the points along the boundary of an image that have the same intensity. It can be used widely in shape analysis, finding the size of the object, image analysis and object detection. It uses the function **cv2.findContours().** We specify the image outputted from dilation (previous step), CV2.RETR_EXTERNAL which retrieves only the extreme outer contours and

CV2.CHAIN_APPROX_SIMPLE parameters helps to trim the diagonal, horizontal and vertical segments except their end points. After extracting the contour from the binary image(Figure 4.5), we need to calculate the contour area to return the area of the text region. It uses the function **cv2.contourArea().** We need to specify the contour 2D vector points and oriented area flag. This function returns a signed area value based on the contour values. After finding the contour area we draw a bounding box with the function **cv2.rectangle().** This function simply draws a filled-up rectangle.

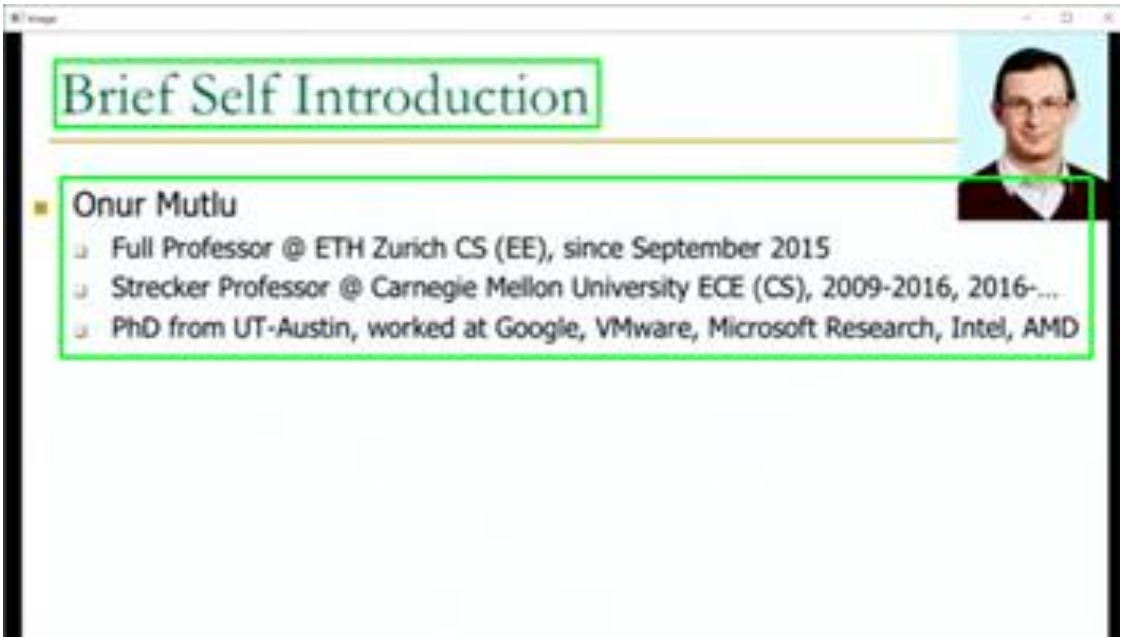

Figure 4.5 After finding contour area and bounding box

4.1.4 Text region refinement

Various image processing can be performed using NumPy (Python Library) functions. NumPy stores images in an array “ndarray” which can be used for the purpose of acquisition and rewriting of pixel values, trimming by slice, and concatenating can be done. Since we are using

OpenCV, Python's OpenCV image values are stored as a ndarray hence these values can be used in the processing of NumPy.

From the previous steps, we have extracted the contour arear of the text region. The region is enclosed using a rectangle thick line box. Using OpenCV we can find the ndarray of the text region which then can be used for the purpose of NumPy slicing. It can be done from the extracted ndarray values which constitute the rectangular region for the purpose of text region refinement. Now this region becomes the ROI "Region of interest" (Figure 4.6) which is the Portion of the image that can be filtered or can perform some other operations.

Figure 4.6 Result after NumPy Slicing

4.1.5 Text extraction

Tesseract is an open source text recognition engine. Tesseract was considered as one of the most accurate open source OCR engines. It supports a variety of languages and is used to recognize text from a large document, or it can also be used to recognize text from an image of a single text line. From paper [36] "*Tesseract is described as a pipeline based architecture which*

consists of the following sequential steps: preprocessing providing a binary threshold, determining the connected components and connections between them, character recognition and character aggregation to form words, lines, paragraph and finally solving the problem of detecting small capitals". By this way we can extract all the text that appears in the video lecture which helps us to gather all the possible metadata which can be used for the purpose of video indexing.

4.2 Automatic Speech recognition

We have used the speech recognition service provided by the Microsoft Azure service to generate the speech transcription. Video indexer API is a toll that allows web apps to make request to the video indexer. After we upload the video into the Microsoft azure, the transcript will be ready in few minutes. As soon the transcript is readily available, we can download the transcript and use it for the purpose of video search. This video transcript plays a major role in video indexing and search. The csv file that is generated from the Microsoft azure cloud consists of three main columns. They are start time of the sentence, end time of the sentence and then the actual sentence. Since it automatically provides the start and the end time, it will be easy for the students to do a search in the video and can also see the time frame where the professor has talked about the specific topic.

4.3 Combined Model

Now we have the metadata extracted from the lecture videos using OCR and ASR Technology. After we have metadata collected it is used for the purpose of video indexing and video search.

4.3.1 Video Indexing

We make use of JavaScript plugins and html to create markers on the video so that it will be easy to navigate to specific topics. The markers can have a title which is extracted from the topic from each slide. From Figure 4.7 the slide title contents are passed as parameters in video markers plugins.

```
//load markers
video.markers({
  markers: [
    {time: 9.5, text: "Memory Systems"},
    {time: 105,  text: "Brief Self Introduction"}

  ]
});
```

Figure 4.7 Video markers plugin

4.3.2 Video Search

From all the possible source of data collection with the availability of metadata we can create a search platform for the students to search for specific topic. The search happens with the metadata that are collected from the video text and the voice transcription of the professor. Figure 4.8 explains the output of the video search results. The student can search for a specific topic or keyword. The search method returns the text that appears from the video and the speech recognition of the professor.

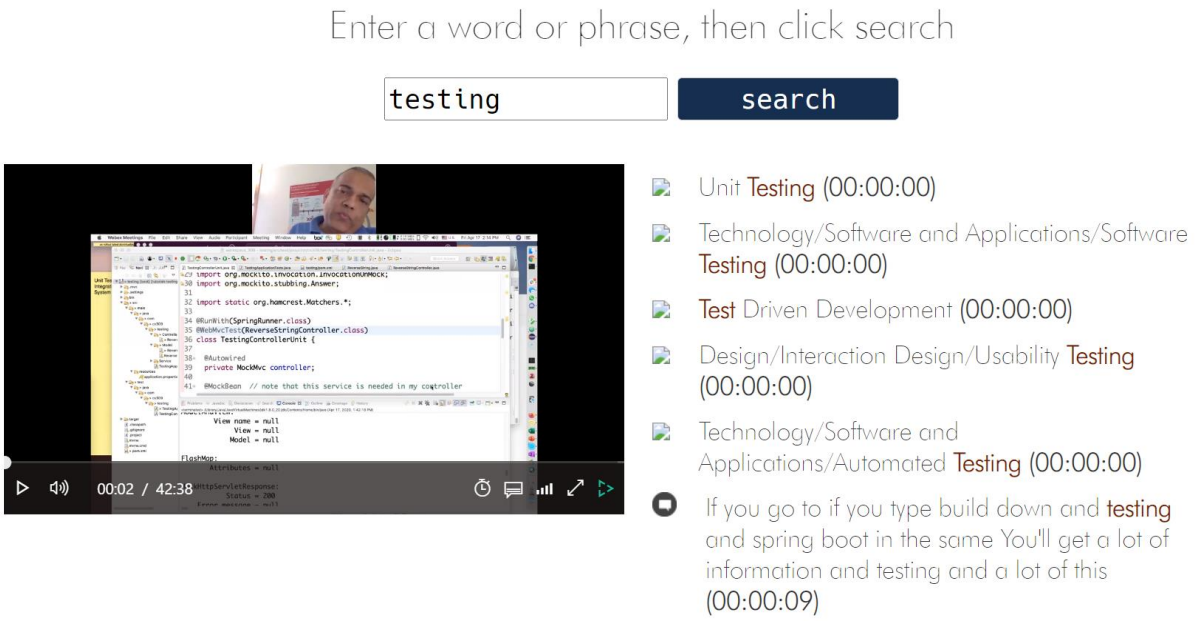

Figure 4.8 Search results

4.3.3 Comparison

The time complexity of this model depends on the size of the lecture video. We tested our model with different sizes of the video. For a 10-minute video the time that it took for processing was half (5 minutes) and increased proportionally based on the total size of the lecture video. We also faced some performance issue of the CPU when we run this model. Hence the size of the video depends on the performance of the CPU. The larger the size of the video, the time it took was increasing. High end GPU units are recommended when the size of the lecture video is increasing exponentially.

CHAPTER 5. CONCLUSION

5.1 Overview of Research

We presented the video indexing and search for lecture videos that are available to students and universities. Unlike the previous methodologies, this system transcribes the video and allows the user to search for specific topics or concepts. We experiment our approach with the lecture videos with different structures of the videos. We conclude that our approach is more efficient when compared to other existing approaches.

5.2 Future Work

Consider other available transcription platforms which are more efficient than our approach. Improve the segmentation process based on collecting bookmarks from students for better performance and more accurate segmentation. Extract keywords or tags from the lecture video which might help the students to get better knowledge about the video without playing it.

REFERENCES

[1] R. F. Kizilcec, K. Papadopoulos, and L. Sritanyaratana, “Showing face in video instruction: effects on information retention, visual attention, and affect”, in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems CHI 2014, April 26–May 1, 2014, Toronto, Ontario, Canada.

[2] M. Pedrotti and N. Nistor, "Online Lecture Videos in Higher Education: Acceptance and Motivation Effects on Students' System Use," *2014 IEEE 14th International Conference on Advanced Learning Technologies*, Athens, 2014, pp. 477-479. doi: 10.1109/ICALT.2014.141.

[3] M. Malchow, M. Bauer and C. Meinel, "Enhance learning in a video lecture archive with annotations," *2018 IEEE Global Engineering Education Conference (EDUCON)*, Tenerife, 2018, pp. 849-856. doi: 10.1109/EDUCON.2018.8363319

[4] S. Anand, S. Chatterjee and K. Bijlani, "Pedagogy Experiments with Recorded Video Lectures," *2014 IEEE Sixth International Conference on Technology for Education*, Clappana, 2014, pp. 193-194. doi: 10.1109/T4E.2014.43.

[5] E. L Glassman, J. Kim, A. Monroy-Hernández, and M. Ringel Morris, "Mudslide: A spatially anchored census of student confusion for online lecture videos", *in Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, New York, NY, 1555–1564. doi: 10.1145/2702123.2702304.

[6] TED ed lessons worth sharing, https://ed.ted.com/

[7] J. Kay, P. Reimann, E. Diebold, and B. Kummerfeld, “MOOCs: so many learners, so much potential ...”, in IEEE Intelligent Systems, vol. 28, no. 3, May 2013, pp. 70-77.

[8] J. L. Martín Núñez, E. Tovar Caro and J. R. Hilera González, "From Higher Education to Open Education: Challenges in the Transformation of an Online Traditional Course," in *IEEE Transactions on Education*, vol. 60, no. 2, pp. 134-142, May 2017. doi: 10.1109/TE.2016.2607693

[9] T. Nagai, T. Toyota, T. Nagoya, K. Nishizawa and M. Imai, "Implementation of high-definition lecture recording system for daily use," *2013 IEEE Global Engineering Education Conference (EDUCON)*, Berlin, 2013, pp. 520-525. doi: 10.1109/EduCon.2013.6530155.

[10] W. JiuHong, W. LiPing, L. MengYang and W. YouWei, "Advantages and Deficiencies of the Automated Multimedia Lecture Recording System in Lecture Video Production," *2009 International Forum on Computer Science-Technology and Applications*, Chongqing, 2009, pp. 271-273. doi: 10.1109/IFCSTA.2009.306.

[11] E. Baralis, L. Cagliero, L. Farinetti, M. Mezzalama and E. Venuto, "Experimental Validation of a Massive Educational Service in a Blended Learning Environment," *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, Turin, 2017, pp. 381-390. doi: 10.1109/COMPSAC.2017.123..

[12] H. Mori, H. Tanaka, Y. Hori, M. Otani and K. Watanabe, "Development of Lecture Videos Delivery System using HTML5 Video Element," *2013 Eighth International Conference on Broadband and Wireless Computing, Communication and Applications*, Compiegne, 2013, pp. 557-559. doi: 10.1109/BWCCA.2013.96

[13] J. Koh, T. Chu and G. C. Lee, "Supporting In-Class Learning with Asynchronous and Autonomous Viewing of Near Real-Time Lecture Videos," *2014 International Conference on Teaching and Learning in Computing and Engineering*, Kuching, 2014, pp. 141-142. doi: 10.1109/LaTiCE.2014.32.

[14] H. Z. Abidin, H. Hussin, M. I. M. Ali, M. Muhamad and Y. Husaini, "Online video lecture series for digital logic fundamental courses blended learning," *2017 IEEE 9th International Conference on Engineering Education (ICEED)*, Kanazawa, 2017, pp. 228-232. doi: 10.1109/ICEED.2017.8251198.

[15] M. Bauer, M. Malchow and C. Meinel, "Improving access to online lecture videos," *2018 IEEE Global Engineering Education Conference (EDUCON)*, Tenerife, 2018, pp. 1161-1168. doi: 10.1109/EDUCON.2018.8363361.

[16] Y. Kometani, T. Furuta and T. Akakura, "Video Bookmarking for Learner Support in Blended Learning: Selection of Appropriate Keywords for Efficient Review of Lecture Video," *2011 IEEE 11th International Conference on Advanced Learning Technologies*, Athens, GA, 2011, pp. 585-586. doi: 10.1109/ICALT.2011.176.

[17] H.-C. Shih and C.-L. Huang, “Content-based multi-functional video retrieval system,” in *Proc. IEEE Int. Conf. Consum. Electron.*, Jan. 2005, pp. 383–384.

[18] S. Mansouri, M. Charhad, A. Rekik and M. Zrigui, "A Framework for Semantic Video Indexing Using Textual Information," *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, Lviv, 2018, pp. 107-110. doi: 10.1109/DSMP.2018.8478609

[19] I. Aljarrah and D. Mohammad, "Video content analysis using convolutional neural networks," *2018 9th International Conference on Information and Communication Systems (ICICS)*, Irbid, 2018, pp. 122-126. doi: 10.1109/IACS.2018.8355453

[20] S. Wang and Q. Ji, "Video Affective Content Analysis: A Survey of State-of-the-Art Methods," in *IEEE Transactions on Affective Computing*, vol. 6, no. 4, pp. 410-430, 1 Oct.-Dec. 2015. doi: 10.1109/TAFFC.2015.2432791

[21] H. Shih, "A Survey of Content-Aware Video Analysis for Sports," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 5, pp. 1212-1231, May 2018. doi: 10.1109/TCSVT.2017.2655624.

[22] T. Tuna, J. Subhlok and S. Shah, "Indexing and keyword search to ease navigation in lecture videos," *2011 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, Washington, DC, 2011, pp. 1-8. doi: 10.1109/AIPR.2011.6176364.

[23] H. Yang and C. Meinel, "Content Based Lecture Video Retrieval Using Speech and Video Text Information," in *IEEE Transactions on Learning Technologies*, vol. 7, no. 2, pp. 142-154, April-June 2014. doi: 10.1109/TLT.2014.2307305.

[24] L. S. Kate, M. M. Waghmare and A. Priyadarshi, "An approach for automated video indexing and video search in large lecture video archives," *2015 International Conference on Pervasive Computing (ICPC)*, Pune, 2015, pp. 1-5. doi: 10.1109/PERVASIVE.2015.7087169.

[25] Yang, M. Siebert, P. Luhne, H. Sack and C. Meinel, "Automatic Lecture Video Indexing Using Video OCR Technology," *2011 IEEE International Symposium on Multimedia*, Dana Point CA, 2011, pp. 111-116. doi: 10.1109/ISM.2011.26.

[26] V. K. Kamabathula and S. Iyer, "Automated Tagging to Enable Fine-Grained Browsing of Lecture Videos," *2011 IEEE International Conference on Technology for Education*, Chennai, Tamil Nadu, 2011, pp. 96-102. doi: 10.1109/T4E.2011.23H.

[27] A. Balagopalan, L. L. Balasubramanian, V. Balasubramanian, N. Chandrasekharan and A. Damodar, "Automatic keyphrase extraction and segmentation of video lectures," *2012 IEEE International Conference on Technology Enhanced Education (ICTEE)*, Kerala, 2012, pp. 1-10. doi: 10.1109/ICTEE.2012.6208622.

[28] A. Park, T. J. Hazen and J. R. Glass, "Automatic processing of audio lectures for information retrieval: vocabulary selection and language modeling," *Proceedings. (ICASSP '05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, Philadelphia, PA, 2005, pp. I/497-I/500 Vol. 1. doi: 10.1109/ICASSP.2005.1415159.

[29] E. Baidya and S. Goel, "LectureKhoj: Automatic tagging and semantic segmentation of online lecture videos," *2014 Seventh International Conference on Contemporary Computing (IC3)*, Noida, 2014, pp. 37-43. doi: 10.1109/IC3.2014.6897144.

[30] R. R. Shah, Y. Yu, A. D. Shaikh and R. Zimmermann, "TRACE: Linguistic-Based Approach for Automatic Lecture Video Segmentation Leveraging Wikipedia Texts," *2015 IEEE International Symposium on Multimedia (ISM)*, Miami, FL, 2015, pp. 217-220. doi: 10.1109/ISM.2015.18.

[31] M. Furini, S. Mirri and M. Montangero, "Topic-based playlist to improve video lecture accessibility," *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, Las Vegas, NV, 2018, pp. 1-5. doi: 10.1109/CCNC.2018.8319246.

[32] C. Bhatt, A. Popescu-Belis, M. Habibi, S. Ingram, S. Masneri, F. McInnes, N. Pappas and O. Schreer, Multi-factor Segmentation for Topic Visualization and Recommendation: the MUST-VIS System," *MM '13 Proceedings of the 21st ACM international conference on Multimedia*, Barcelona, Spain — October 21 - 25, 2013, pp. 365-368. doi: 10.1145/2502081.2508120.

[33] J. Glass, T. J. Hazen, S. Cyphers, I. Malioutov, D. Huynh and R. Barzilay, "Recent Progress in the MIT Spoken Lecture Processing Project," *INTERSPEECH 2007, 8th Annual Conference of the International Speech Communication Association*, Antwerp, Belgium, August 27-31, 2007.

[34] I. Malioutov and R. Barzilay, "Minimum cut model for spoken lecture segmentation," *ACL-44 Proceedings of the 21st International Conference on Computational Linguistics*, Sydney, Australia — July 17 - 18, 2006. doi: 10.3115/1220175.1220179.

[35] Luca Cagliero, Lorenzo Canale, Laura Farinetti and Politecnico di Torino, “VISA: A supervised approach to indexing video lectures with semantic annotations”, *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, Torino, Italy. DOI 10.1109/COMPSAC.2019.00041